2

Multi-Aperture Vision System Coupled To Neural Network Processors

Progress Report

December 15, 1992

To

Office of Naval Research
Department Of The Navy

From

Federal Electro-Optics, Inc.
3873 Central Avenue
Memphis, Tn 38111
(901) 458-3017

Contract Number N00014-92-C-0167

Scientific Officer

Keith Bromley, Ph.D.

425 931 2788

INTRODUCTION

The following report is submitted in partial fulfillment of *Contract Number N00014-92-C-0167* to satisfy Item 0001AB of Section B, page 2 of the contract agreement. It represents Federal Electro-Optics' (FEO) first three months of work on the contract. FEO has made substantial progress in these first three months. The project is currently on schedule.

As outlined in the original proposal, FEO intends to determine how effectively a neural network coupled to a multiple aperture vision system (MAVS) can be applied to the tracking task. Over the course of the contract FEO will perform seven (7) experimental simulations to evaluate the proposed tracking systems' performance. The October 15, 1992 report discussed the modeling of the first three (3) configurations. The following report discusses the results of the implementation of these first three experiments on the Silicon Graphics IRIS Indigo workstation. Additionally, FEO has completed the optical modeling of experiments four through seven and completed the C++ coding of experiments four and five.

FEO has enclosed a DAT tape containing executable code and source code for experiments #1 through #3. The programs can be loaded onto a Silicon Graphics workstation using the *tar* tape facility. A description of the operation of each program is provided below. Once begun, operation of the executable programs is self-explanatory.

EXPERIMENT ONE: IMPLEMENTATION

Review: Experiment number one was the simplest of all the simulations and consisted of a two-dimensional situation whereby one angular target position could be determined. A single lens and five detectors were implemented to determine a point source position placed at random angular positions. For the first three experiments, this configuration defines an *eyelet.*

Optical Model: As shown in Figure 1, the optical model simulates a gradient index lens (GRIN) coupled to five detectors. The following assumptions are made for this model.

1. The lens has a gradient index intensity characteristic.
2. There are five detectors with negligible space between elements.
3. The point source is located at ∞ , thus, the angular displacement, θ, will be used instead of the linear displacements *x,z or d.* The reason for this initial assumption derives from the ambiguities associated with a single line-of-sight.
4. The maximum voltage from any detector is 1-Volt while the minimum is 0-Volts. Although this may not be easily done in hardware, we will implement software programming in a Phase II effort to normalize the detector voltage. This step facilitates the neural network algorithm requirements.
5. The lens will be given an $\approx$ 15° field-of-view (FOV), or about 250 milliradians, which is consistent with several commercially available SELFOC lenses.

C++ Coding: The training data for experiment one consisted of random angles between plus and minus 125 mrad with the corresponding voltages from the detectors as determined by the optical model. A histogram of the data is shown in figure 2. The random number generator in the program had a uniform distribution and as seen in the figure, the data has a uniform distribution as well.

A-1

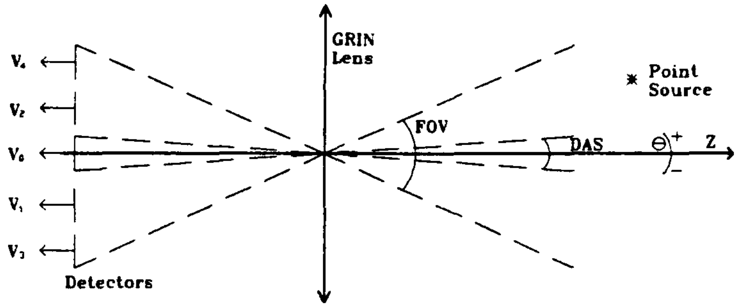


Figure 1. Experiment #1 Configuration.

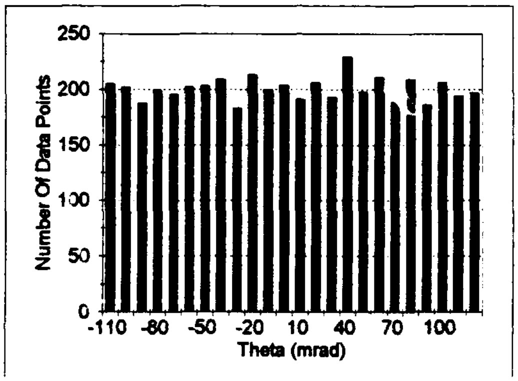


Figure 2. Experiment #1: 1-D Histogram of Neural Network Training Data.

Neural Network Analysis: **Experiment # 1 Network Results**: As discussed in the October report, FEO originally anticipated using a network with only one hidden layer. FEO found that this configuration yielded poor results, therefore, neural networks with *two* hidden layers were implemented. Two networks were trained using the random data generated from experiment 1. Both networks used two hidden layers, but the number of PEs in the two networks is different. The first network used three PEs in the first hidden layer and one in the second hidden layer. The second network has five PEs in the first hidden layer and three PEs in the second hidden layer. Since the overall performance of the second network was considerably better than that of the first, the second architecture was used Figure 3). Statistics for the network are given below:

Total Number of Trials 500
Number of hits 500
Number of near misses 0
Number of misses 0
Mean Angular error: -1.685037 radians.
Standard Deviation of Angular error: 10.159072%

The number of trials indicates the number of sets the network was *tested* on. A hit is defined as a network response that would excite the same detector as the desired response. A near miss is a network response that would activate a detector adjacent to the one excited by the desired response. A miss is anything else. The standard deviation is given in percentage of the FOV.

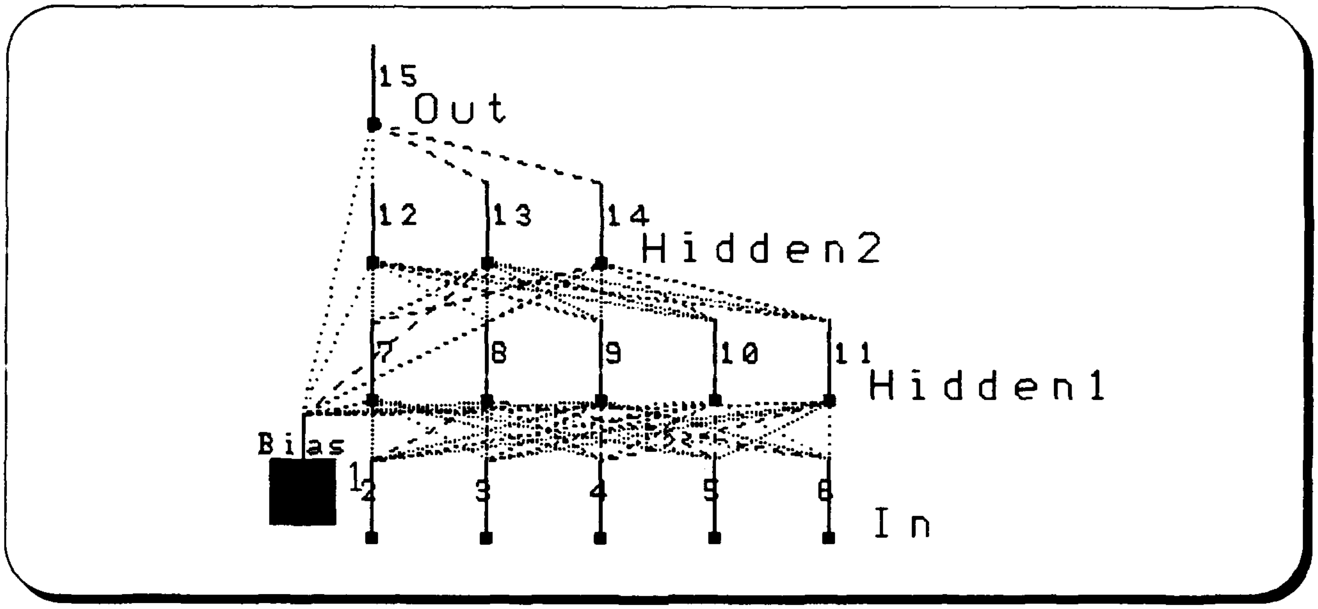


Figure 3. Experiment #1: Final Neural Network Topology.

A histogram of the magnitudes of the errors is shown in Figure 4. The errors greater that 0.2 are caused by fringe effects, that is, the detector voltage for sources on the outside edge of the FOV is extremely small (below noise level).

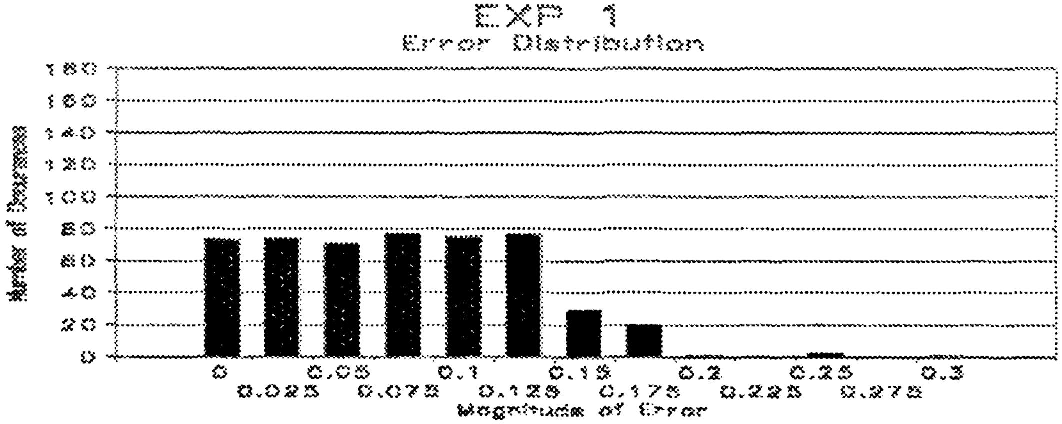


Figure 4. Experiment #1: Histogram Distribution of Neural Network Test Data Errors.

The next set of figures (5 & 6) represent the error vs. position. The alternating line along the top of each graph indicates the spatial extent of the five detectors. *These graphs clearly indicate that there is a relationship between the error and the point source's position on the detector.* FEO believes the symmetry is due to the symmetry of the GRIN lens and the shape due to the network's attempt to map the GRIN lens transfer function. Note that the neural network response in Figure 6 is beginning to approach the expected linear response, i.e., the slope of the line is changing, particularly away from the center detector. When the number of detectors increase, as in Experiment #3, this *fit* becomes even better (see Figure 20).

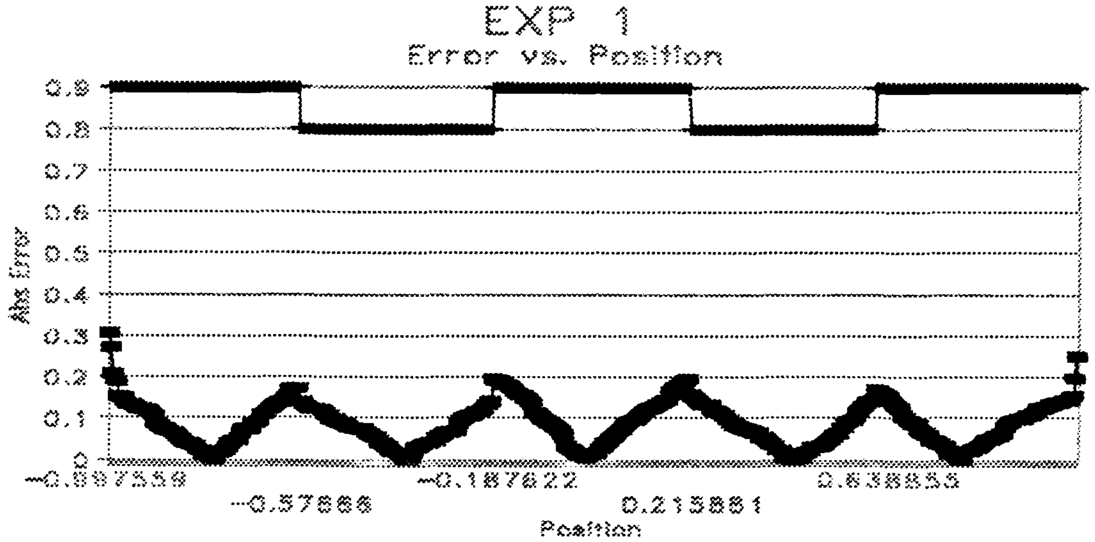


Figure 5. Experiment #1: Absolute error versus Position for Neural Network Test Data.

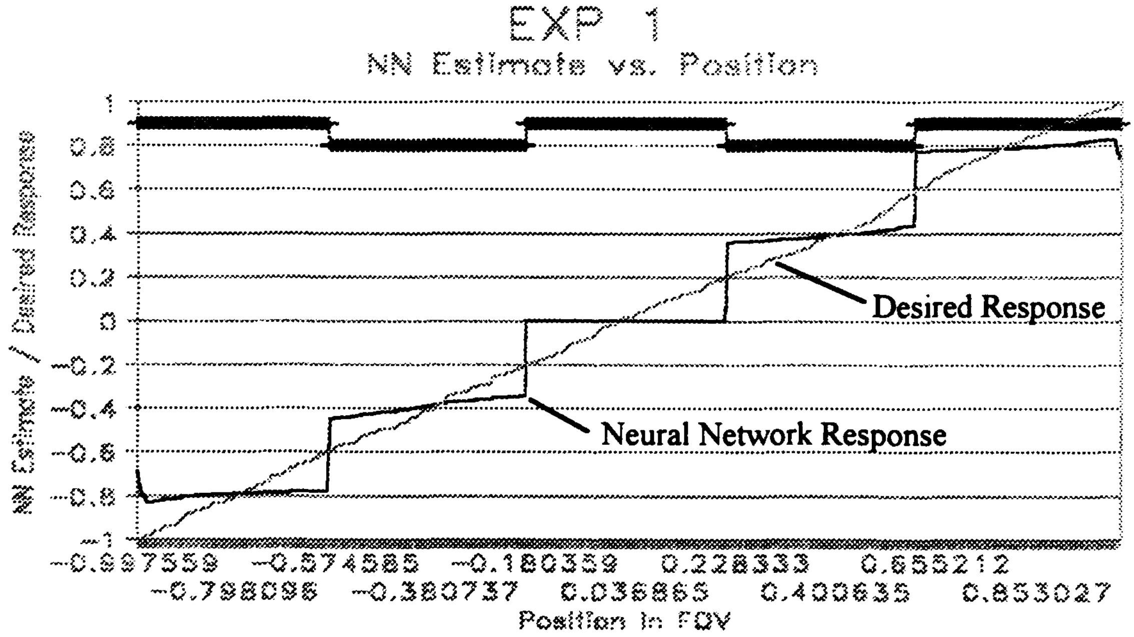


Figure 6. Experiment #1: Neural Estimate versus Actual Point Source Position.

Silicon Graphics Demonstration Program: Our intent was to provide an easy-to-use interface whereby tracking simulations could be run with a minimum knowledge of the MAVS simulation. Once the simulation program is started, the graphical screen appears very similar to that shown in

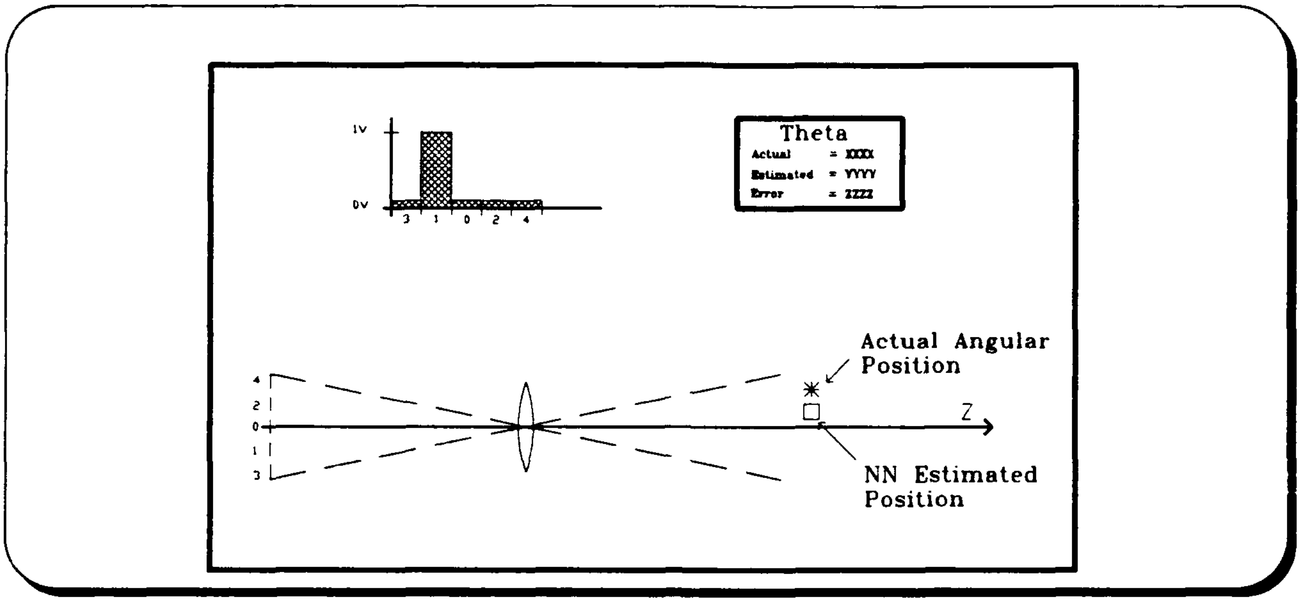


Figure 7. Experiment #1: Silicon Graphics Workstation Screen Graphics.

Figure 7. For this experiment motion is limited to the angular variable, θ, so that the display of the point source and neural network estimate is limited to the vertical direction. The program is totally interactive and is performed in real-time. The point source is moved by using the *mouse* device connected to the Silicon Graphics workstation. The mouse pointer is separate from the *point source graphic*. This allows the user to move the mouse pointer anywhere on the screen (as long as it is to the right of the GRIN lens). As the point source is moved, detector voltages are calculated based on the optical models representing the GRIN lenses. These voltages are submitted to the *trained* neural network. The output of the neural network, which represents the estimate of the angular displacement from the optical axis, is then used to position the *graphical square*. The bars in the upper left hand corner represent the varying voltages from the detectors as the target is randomly positioned. These change in real-time as the point source location is changed. The information in the upper right hand corner represents the *Actual and Neural Network Estimated* target position and the *Error* between the two. The interactive program is very instructional. The curves described in Figure 5 and 6, and the information they represent, are readily observable in the simulation.

EXPERIMENT TWO: IMPLEMENTATION

Review: Experiment two basically expanded experiment one into a multiple aperture simulation. Here, three of the eyelets, as defined in experiment one, are aligned such that their optical axes are in parallel. Once again, the point source will be randomly positioned to give different angular responses to the three eyelets. For this case, the random orientation code also allows the point source to be positioned nearer to the eyelets (recall that experiment one positioned the target at $z = \infty$). Note that this configuration has the potential to provide passive ranging.

Optical Model: As this experiment is very similar to the first simulation, many of the assumptions made there will hold here.

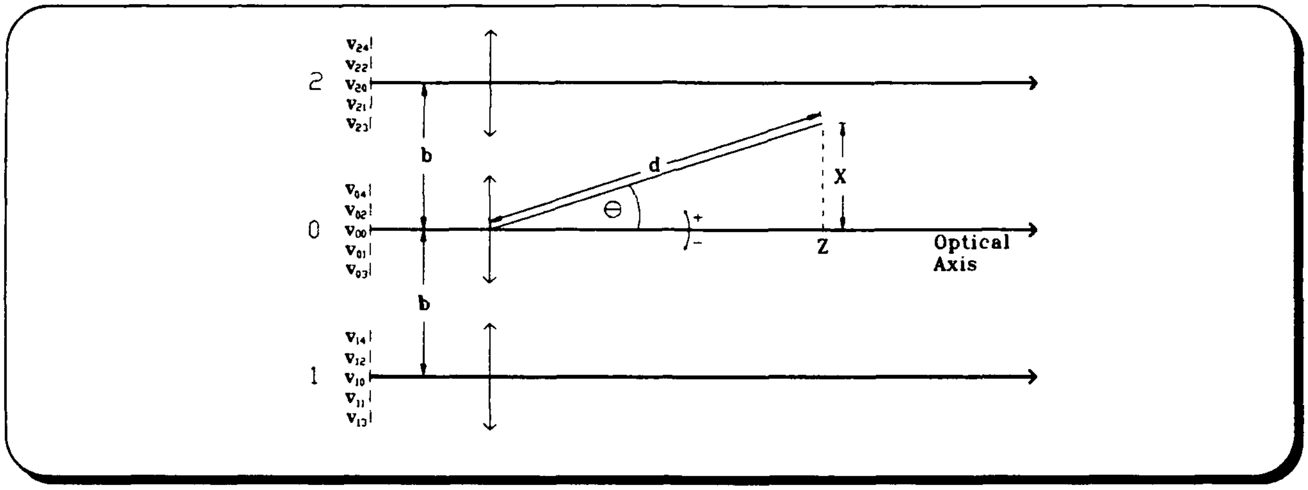


Figure 8. Experimental Configuration Two: Three Eyelets With Parallel Optical Axes.

1. For this experiment the variables to be determined by the neural network are *x and z.*
2. When $z = \infty$, this experiment is basically identical to the first since $V_{04} = V_{14} = V_{24}$, etc., etc. Therefore, this experiment will also include positions when $z \neq \infty$.
3. Comparison of the performance between experiment one and two. Determine which provides better accuracy.
4. Determination of the accuracy of *x and z.*
5. Comparison of off-axis changes to accuracy of determining *z.*

C++ Coding: Generation of the training data for experiment two required that certain assumptions be made prior to coding. The point source for experiment one was at infinity and since the optical models are functions of θ (angle from the optical axis) only a single random number needed to be generated. Experiment two had the point source at a non-infinite distance and therefore, two linear displacements as it's variables for location. In order to have meaningful values for the point source angular location, it was required to restrict the distance of the point source from the lenses to a "reasonable" value. It was decided to normalize the data to the diameter of the lenses and set "b" (the lens separation) equal to one. This meant that there was no separation between the sides of the lenses.

Another decision was made to restrict the data to points in object space where at least two detectors were on at any given location. This was based upon the difficulty of determining an X-Y location with only a single angle. It was then empirically determined that a distance "Z" of 30 "units" yielded acceptable training data that was a good mix of locations with two and three detectors on at a single time.

Neural Network Analysis: **Experiment # 2 Network Results**: The neural network model chosen for experiment two is similar to the network developed for experiment one. Both models used the "data funnel", that is, all layers incorporate fewer PE's than their input layer, with the possible exception of the final output layer. The final neural network used for experiment #2 is shown in Figure 9. Note that the middle layers are identical to those in experiment #1. Note also that not all 15 inputs are shown in the report figure (this was done to enlarge the graphics to a readable scale).

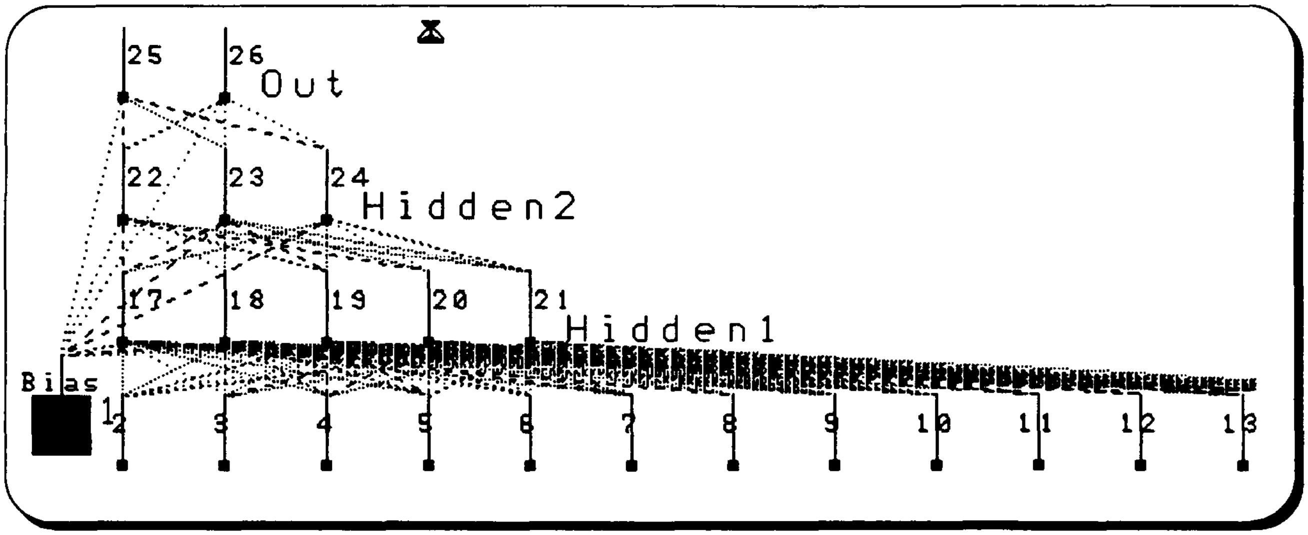


Figure 9. Experiment #2: Neural Network Topology.

The two outputs of experiment two are an *offset* from the optical axis and a *range* from the center of the detector array. The NN was also trained and tested using techniques similar to those employed in experiment one. A "train A, test B" strategy was employed in both cases. 5000 records were used in each training set and 500 in each test set. The fully trained network performed reasonably well, having a normalized mean error of 0.0076 for the offset and 0.0084 for the range. The standard deviations for the offset and range are 9.65% and 11.1%, respectively. A histogram of the error distribution is included in figure 10.

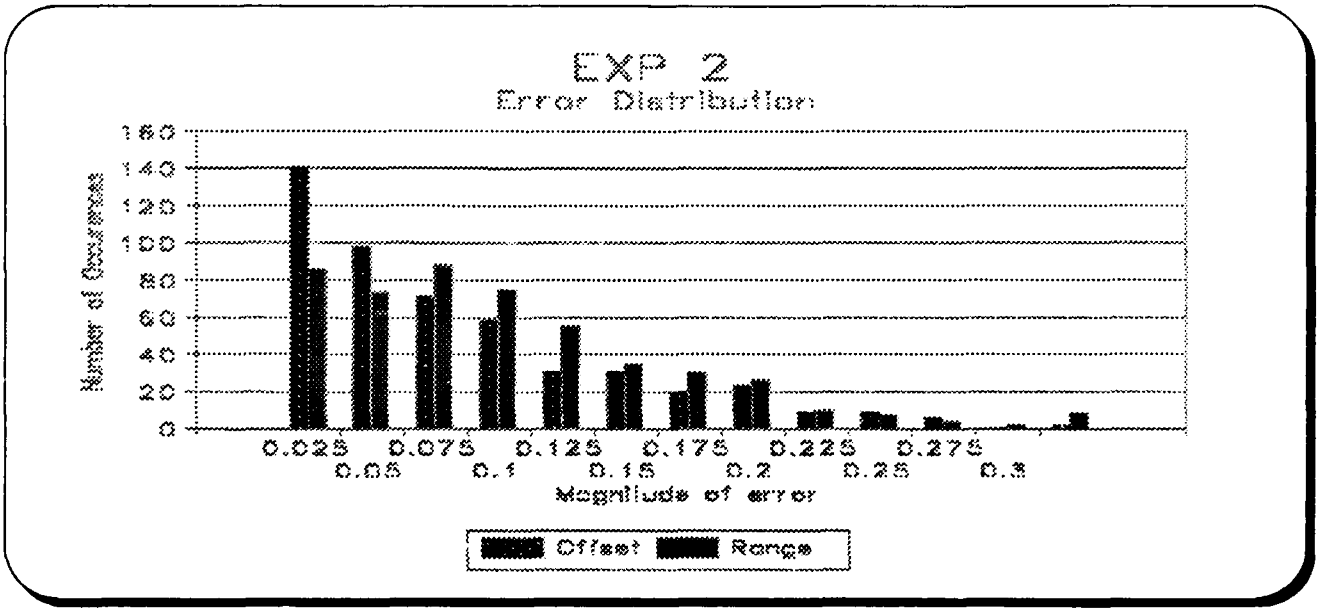


Figure 10. Experiment #2: Histogram Distribution of Neural Network Test Data Errors.

The magnitude of the error associated with the offset parameter was plotted against position in the FOV, and as in experiment 1 shows some position dependence. The relationship is not as pronounced, however. A graph is also included that relates the normalized range to its error. It appears that the error has two zero crossings, one around 0.25 of the full range, and the other around 0.75.

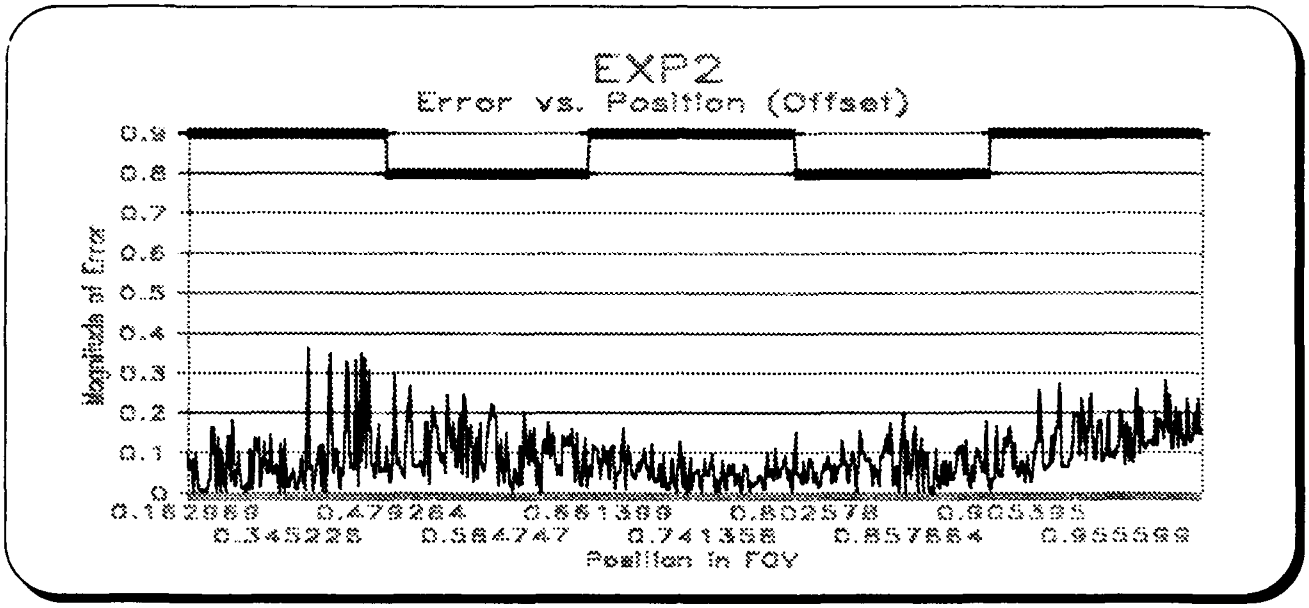


Figure 11. Experiment #2: Error versus Offset Position.

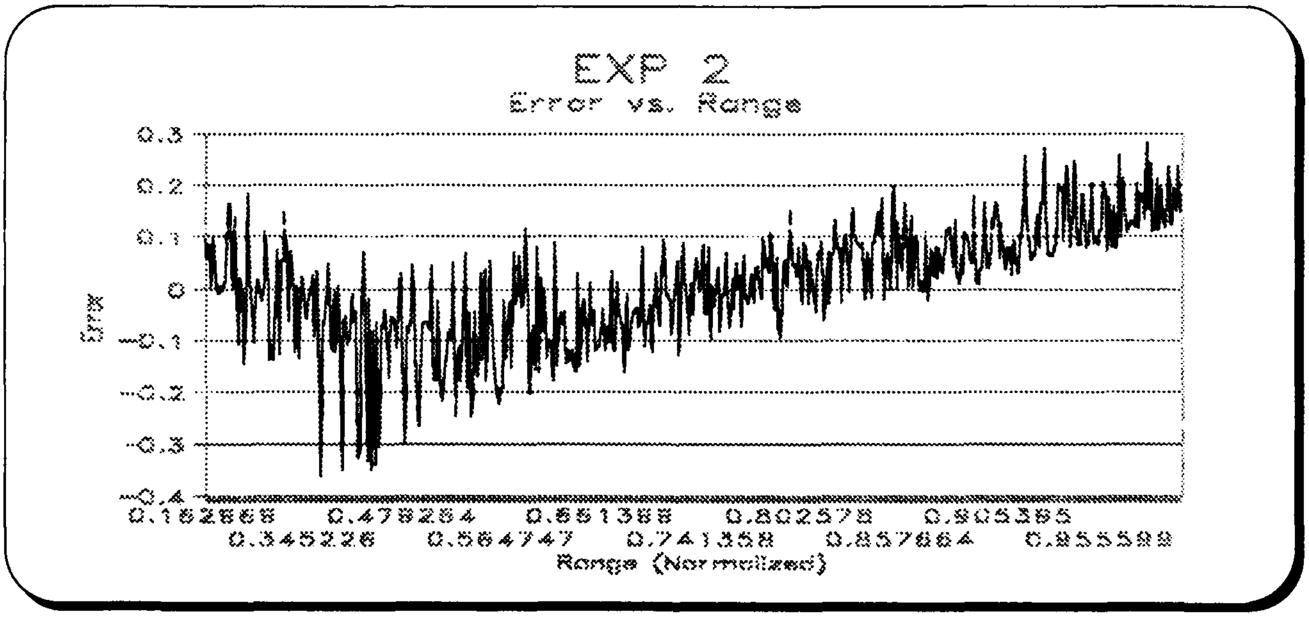


Figure 12. Experiment #2: Error versus Range.

The network's response was also plotted against the desired response for both of the output parameters. As with experiment one's angular output, the network tends to estimate the offset parameter somewhat in the middle of each detector to minimize its error. The tendency is far less noticeable in experiment two, however, with the network's estimates closer to the desired response in the central portion of the FOV (Figure 13). The plot of the network's response versus range indicates that the network tracks the desired response reasonably well until about 0.8 of the normalized range (actually, the estimated range is rather erratic - the average of the estimate does track the range of the point source reasonably well). After this point it appears that the network's estimate simply oscillates around 0.8 (Figure 14). These results were better than FEO anticipated considering range is being determined by three one-dimensional, in-line detector systems with optical axes *parallel* to each other. FEO expects the range estimates will improve with experiments #4 through #6 when two-dimensional eyelets are implemented.

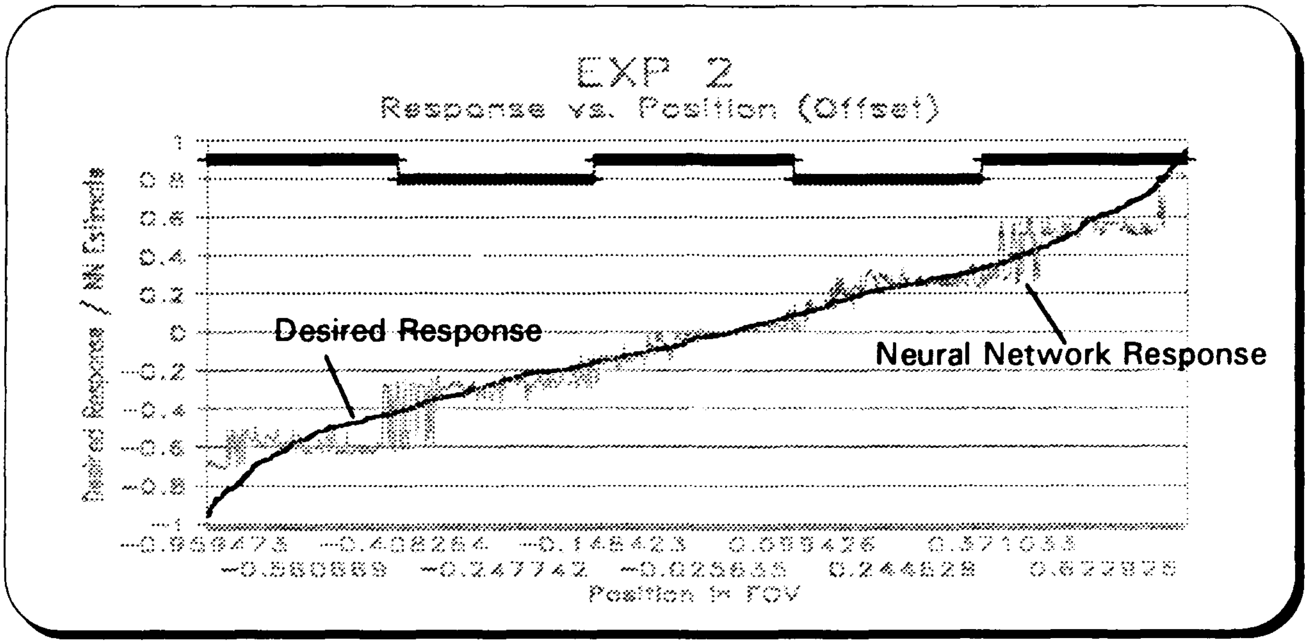


Figure 13. Experiment #2: Neural Network Estimate versus Actual Point Source Position Offset.

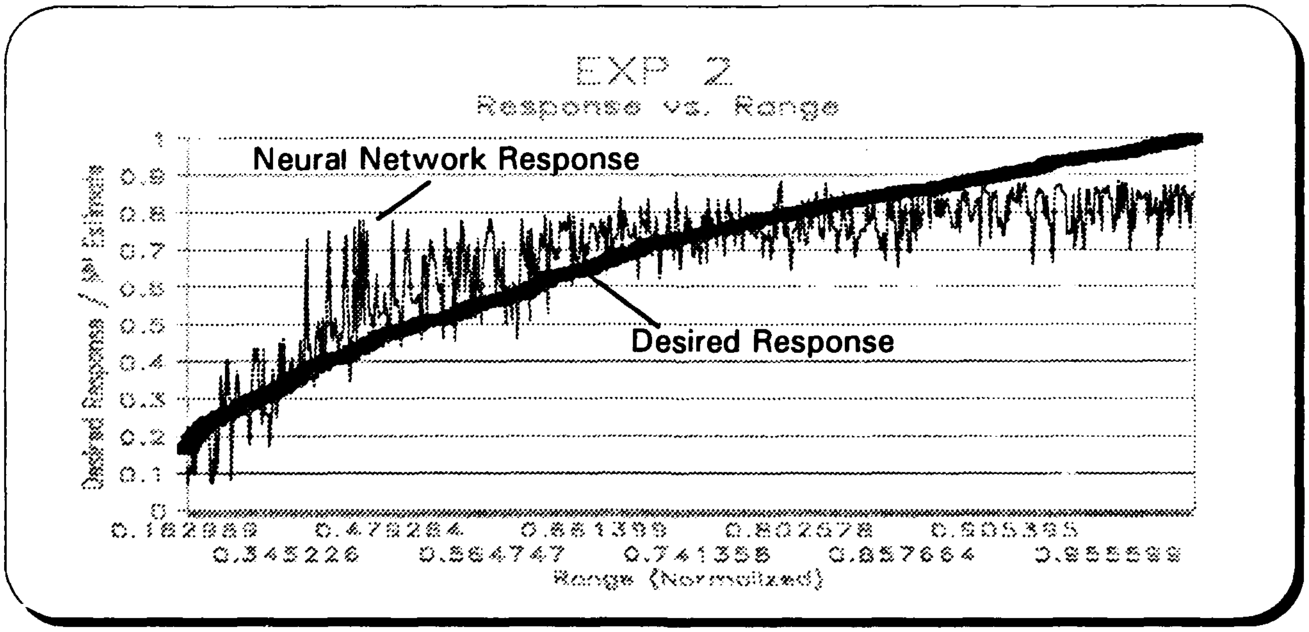
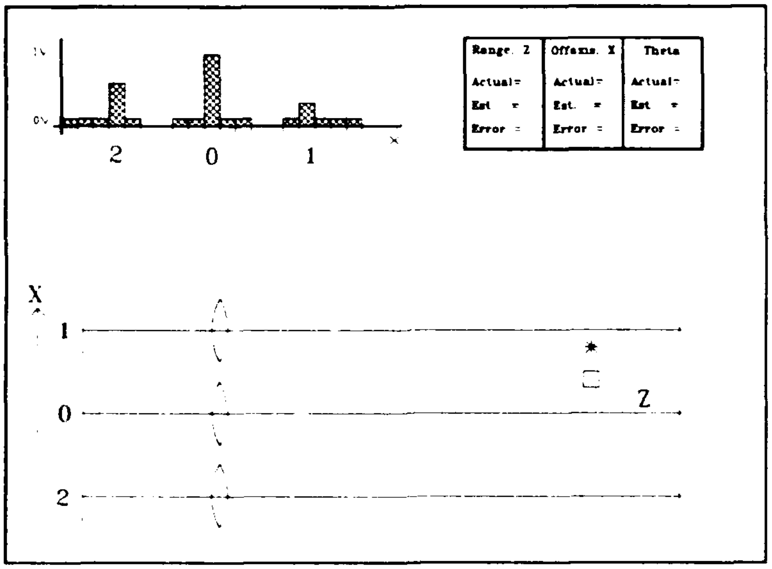


Figure 14. Experiment #2: Neural Network Estimate versus Actual Point Source Range.

Silicon Graphics Demonstration Program: Once the simulation program is started, the graphical screen appears very similar to that shown in Figure 15. For this experiment the point source may be moved to any point in the sensor system's FOV (i.e., it is not limited to a vertical trajectory as in experiment #1). The program is totally interactive and is performed in real-time. The point source is moved by using the mouse device connected to the Silicon Graphics workstation. The mouse pointer is separate from the point source graphic. This allows the user to move the mouse pointer anywhere on the screen (as long as it is to the right of the three GRIN lens). As the point source is moved, detector voltages are calculated based on the optical models representing the GRIN lenses. These voltages are submitted to the trained neural network. The output of the neural network, which represents the estimate of the offset position (x) from the optical axis and the estimated range distance, is then used to position the graphical square. The bars in the upper left hand corner represent the varying voltages from the detectors as the target is randomly positioned. These change in real-time as the point source location is changed. The information in the upper right hand corner represents the Actual and Neural Network Estimated target position and the Error between the two. The interactive program is very instructional. The curves described in Figure 13 and 14, and the information they represent, are readily observable in the simulation (i.e., the graphical square tracks the x-position well but tends to be erratic when providing the range).



Range. Z	Offaxis. X	Theta
Actual=	Actual=	Actual=
Est =	Est. =	Est =
Error =	Error =	Error =

Figure 15. Experiment #2: Silicon Graphics Workstation Graphics Screen.

EXPERIMENT THREE: IMPLEMENTATION

Review: Experiment number three represents the final two-dimensional eyelet simulation and allows the angle between the three eyelets to be varied. For this experiment the random position generator will again be limited to z = ∞ since the eyelet axes are not parallel.

Optical Model: As shown below, the optical model simulates three eyelets positioned at various optical axis angles. The following assumptions are made for this model.

1. All sensor FOVs are considered to be positioned at the same detection point, i.e., the source is positioned at very large distances from the eyelets.
2. The FOVs overlap by one-half of the sensor FOVs

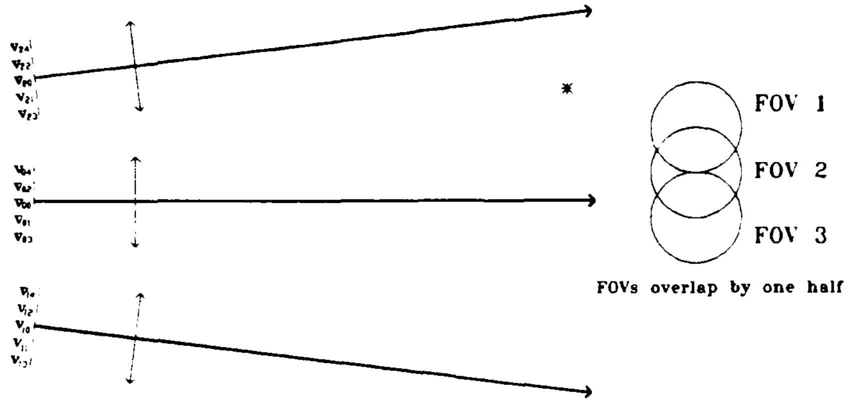


Figure 16. Experimental Configuration Three.

C++ Coding: Experiment three was very similar to experiment #2, however, only a single random number needed to be generated for the angular location of the point source at infinity. As in experiment two, it was decided to limit the training data to locations with at least two detectors on at a time. For this case, and with an angular separation of the lenses of 125 mrad, the data generation becomes identical to experiment one. Figure 17 shows a histogram of the training data for experiment three. As expected, the histogram is identical to the training data for experiment one since the same seed was used for the random number generator.

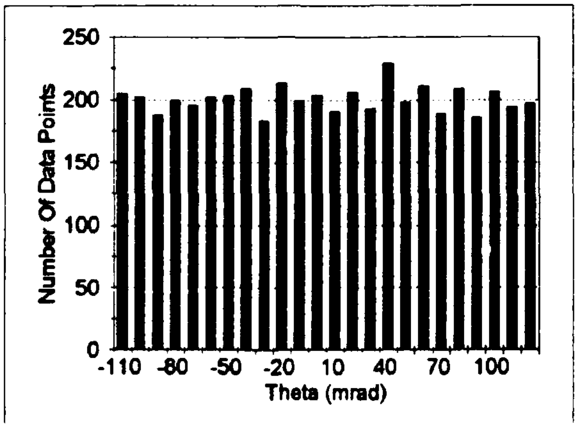


Figure 17. Experiment #3: 1-D Histogram of Neural Network Training Data.

Neural Network Analysis: Experiment #3 Results: The setup of experiment 3 is essentially a three lens expansion of experiment 1. The neural network model chosen for experiment three is identical to that from experiment 1, with the exception of the input layer. See figure 18. Note also that not all 15 inputs are shown in the report figure (this was done to enlarge the graphics to a readable scale).

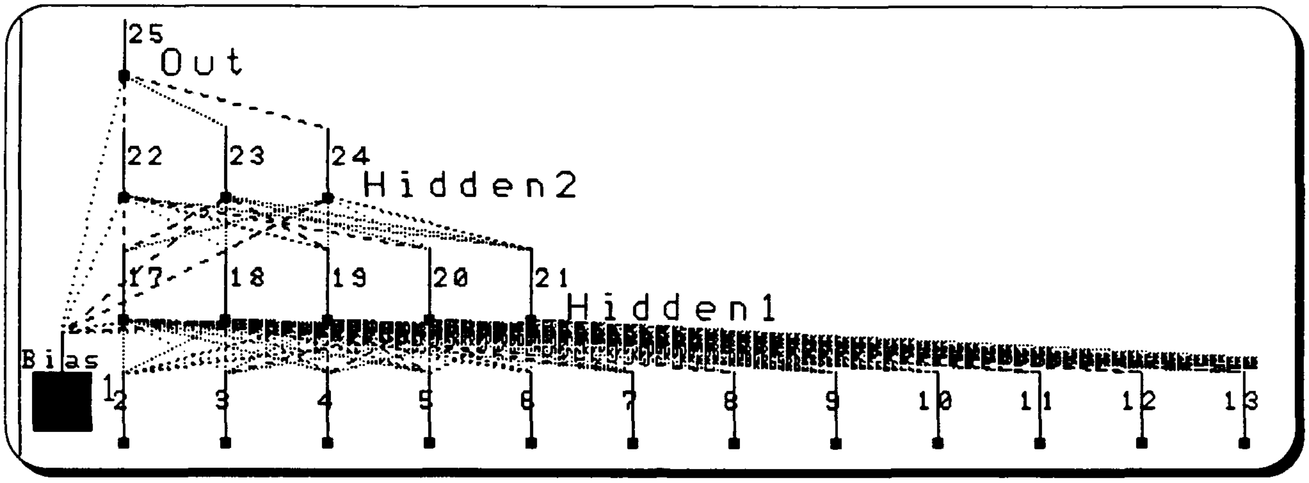


Figure 18. Experiment #3: Neural Network Topology.

The neural network trained for experiment 1 had a normalized average error of -0.00337. The neural network for experiment 3 exhibited a substantial improvement over experiment #1. The mean error for experiment #3 was -0.00082, approximately four times better. The standard deviation of the error also shows a vast improvement - down from 10.16% in experiment #1 to 5.07% in experiment #3, approximately two times better. The enhanced performance is reflected in the error distributions shown in the following figures.

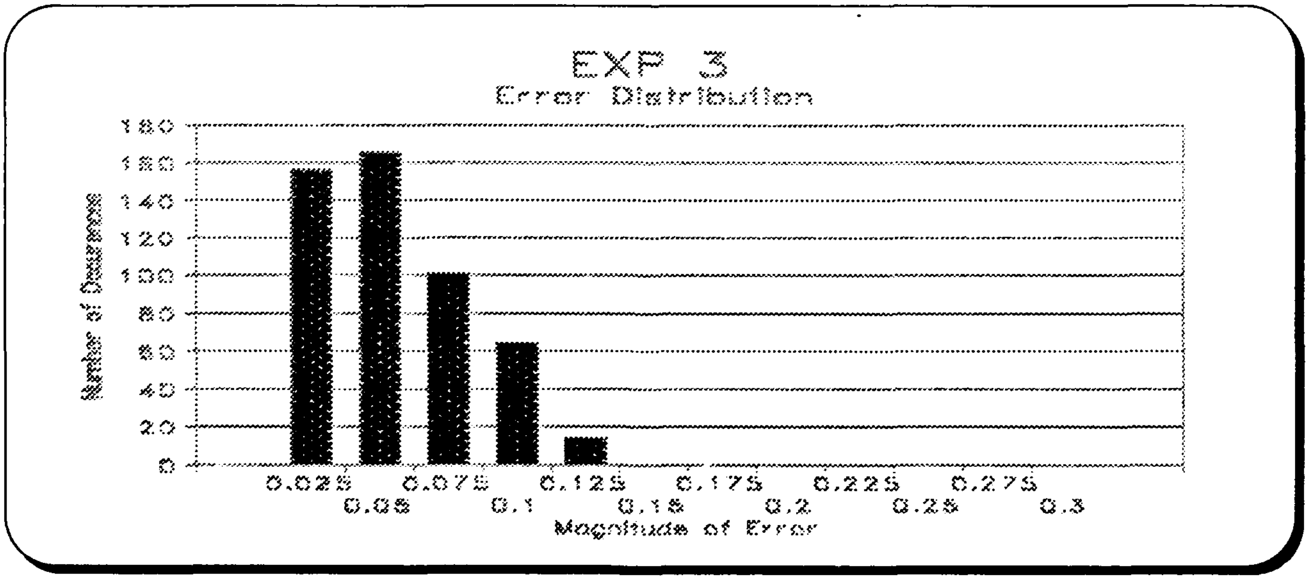


Figure 19. Experiment #3: Distribution of Neural Network Test Data Errors.

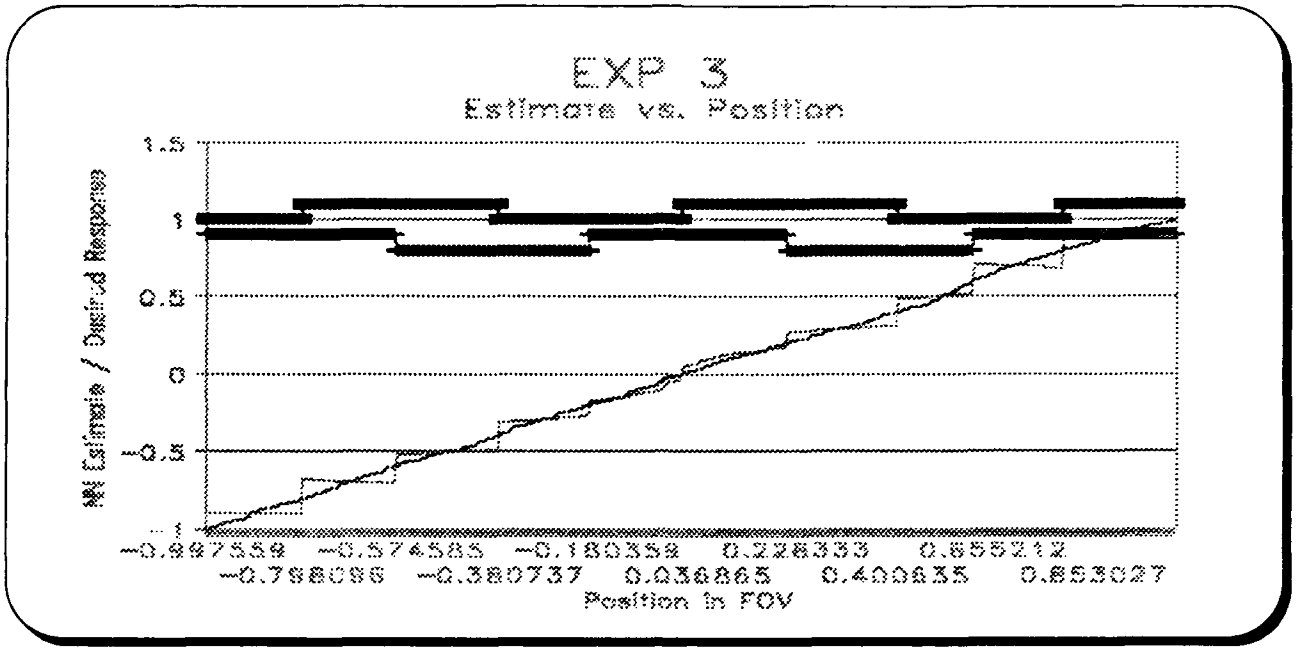


Figure 20. Experiment #3: Neural Network Estimate versus Actual Point Source Location.

An interesting result of the comparison of the two experiments is that it appears the network minimizes its error by estimating the output close to the center of the detector due to each set of inputs. This becomes clear when one studies Figure 6 of Estimate vs Position for experiment #1 and Figure 20 above. The discontinuities correspond to the ends of a given detector, (i.e. the discontinuities of the detector array). The similar graph for experiment three substantiates this theory, with the increased number of discontinuities in the graph still corresponding to the discontinuity in the detector array.

Silicon Graphics Demonstration Program: Once the simulation program is started, the graphical screen appears very similar to that shown in Figure 21. For this experiment motion is limited to the angular variable, θ, so that the display of the point source and neural network estimate is limited to the vertical direction. The program is totally interactive and is performed in real-time. The point source is moved by using the *mouse* device connected to the Silicon Graphics workstation. The mouse pointer is separate from the *point source graphic*. This allows the user to move the mouse pointer anywhere on the screen (as long as it is to the right of the three GRIN lens). As the point source is moved, detector voltages are calculated based on the optical models representing the GRIN lenses. These voltages are submitted to the *trained* neural network. The output of the neural network, which represents the estimate of the angular displacement from the optical axis, is then used to position the *graphical square*. The bars in the upper left hand corner represent the varying voltages from the detectors as the target is randomly positioned. These change in real-time as the point source location is changed. The information in the upper right hand corner represents the *Actual and Neural Network Estimated* target position and the *Error* between the two. The interactive program is very instructional. The curves described in Figure 19 and 20, and the information they represent, are readily observable in the simulation. Note that the *graphical square* tracks the point source noticeably better than in experiment #1.

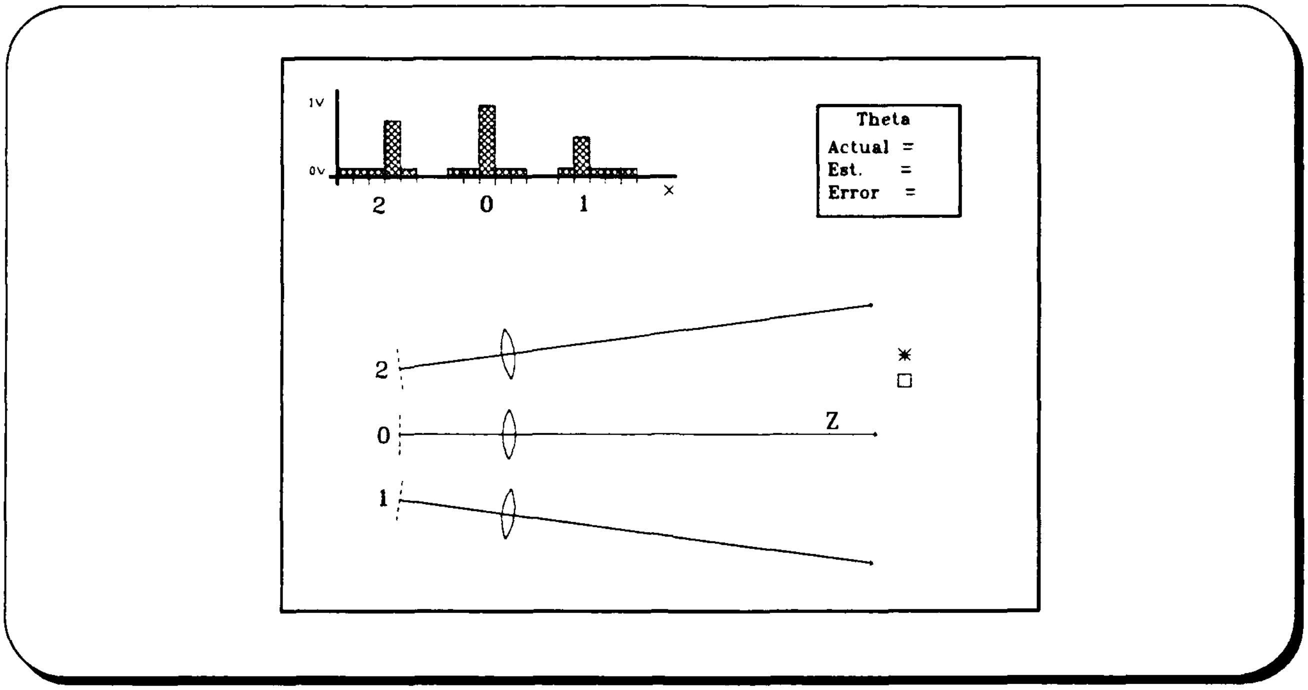


Figure 21. Silicon Graphics Representation of Screen Graphics for Experimental Simulation Three.

EXPERIMENT FOUR: MODELS AND CODING

Experiment number four begins a series of three dimensional imaging investigations. However, this investigation begins with the consideration of a single eyelet as shown in figure 22.

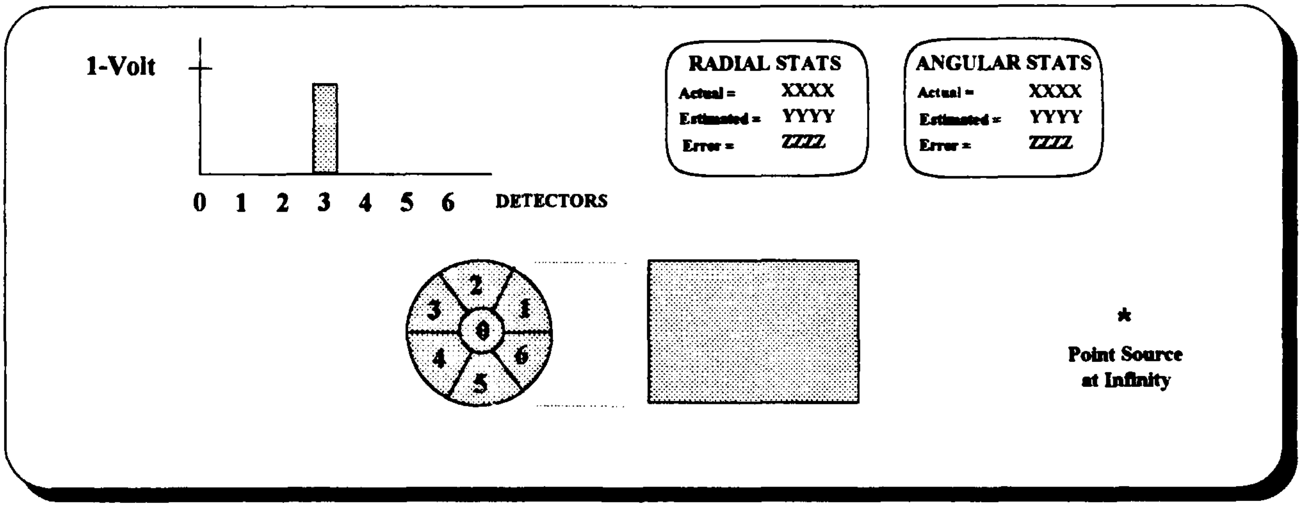


Figure 22. Experiment #4: Silicon Graphics Workstation Screen Graphics.

Consider an eyelet that views a point source located at an infinite conjugate. The eyelet has seven detectors at the image plane as shown. Each detector response will be specified by a point source location where the point source location is completely described by a radial off-axis angle, ϕ, and an angular direction from the 1/6 detector interface, θ, in the counterclockwise image plane direction. The detector voltages are given below:

$$\upsilon_0 = \sqrt{1 - \frac{\phi^2}{(125)^2}}, \qquad \phi \leq 25\,mrad$$

$$\upsilon_1 = \sqrt{1 - \frac{\phi^2}{(125)^2}}, \qquad 25 \leq \phi \leq 125\,mrad,\ 0 \leq \theta \leq \frac{\pi}{3}$$

$$\upsilon_2 = \sqrt{1 - \frac{\phi^2}{(125)^2}}, \qquad 25 \leq \phi \leq 125\,mrad,\ \frac{\pi}{3} \leq \theta \leq \frac{2\pi}{3}$$

$$\upsilon_3 = \sqrt{1 - \frac{\phi^2}{(125)^2}}, \qquad 25 \leq \phi \leq 125\,mrad,\ \frac{2\pi}{3} \leq \theta \leq \pi$$

$$\upsilon_4 = \sqrt{1 - \frac{\phi^2}{(125)^2}}, \qquad 25 \leq \phi \leq 125\,mrad,\ \pi \leq \theta \leq \frac{4\pi}{3}$$

$$\upsilon_5 = \sqrt{1 - \frac{\phi^2}{(125)^2}}, \qquad 25 \leq \phi \leq 125\,mrad,\ \frac{4\pi}{3} \leq \theta \leq \frac{5\pi}{3}$$

$$\upsilon_6 = \sqrt{1 - \frac{\phi^2}{(125)^2}}, \qquad 25 \leq \phi \leq 125\,mrad,\ \frac{5\pi}{3} \leq \theta \leq 2\pi$$

Consider the graphical representation of the eyelet. For the angular radius, ϕ, to be less than 25 milliradians means that the target resides on detector 0. From this point on, the angular radius will be regarded as just *radius*. The angle, θ, of the target with respect to the 1/6 detector interface in radians will be regarded as just target *angle*. The units of radius will be milliradians and the units of angle will be radians. When the target resides on the 0 detector, the signal from the detector does contain radial information, however, target angle will be difficult for the neural network to estimate. Subsequent experiments will overlap non-coaxial detectors with other 0 detectors allowing a better determination of the target angle.

With regard to the other detectors, FEO expects the radial position to be determined to a greater accuracy on detectors 1 through 6 since the parabolic response is more sensitive in these areas.

The angular target position will be determined with an *anticipated* performance of better than $\pi/3$. Again, these accuracies are expected to increase as more eyelets are overlapped. In summary, FEO intends to characterize the response of a single eyelet, but its performance in target location tasks is not expected to be significant. These performance measurements will be more interesting for the multi-aperture cases to follow in experiments #6 and #7.

FEO anticipates using a similar neural network to those used in experiment's #1 through #3 for experiment #4 except that the input will consist of 7 processing elements. The middle layers will remain as is for the first experimental iteration. The identical analyses of the results performed in the first three experiments will be implemented here also. The results of the simulation will be compared to the first three experiments and subsequently reported. Additionally, the simulation interface will be identical to the first three experiments.

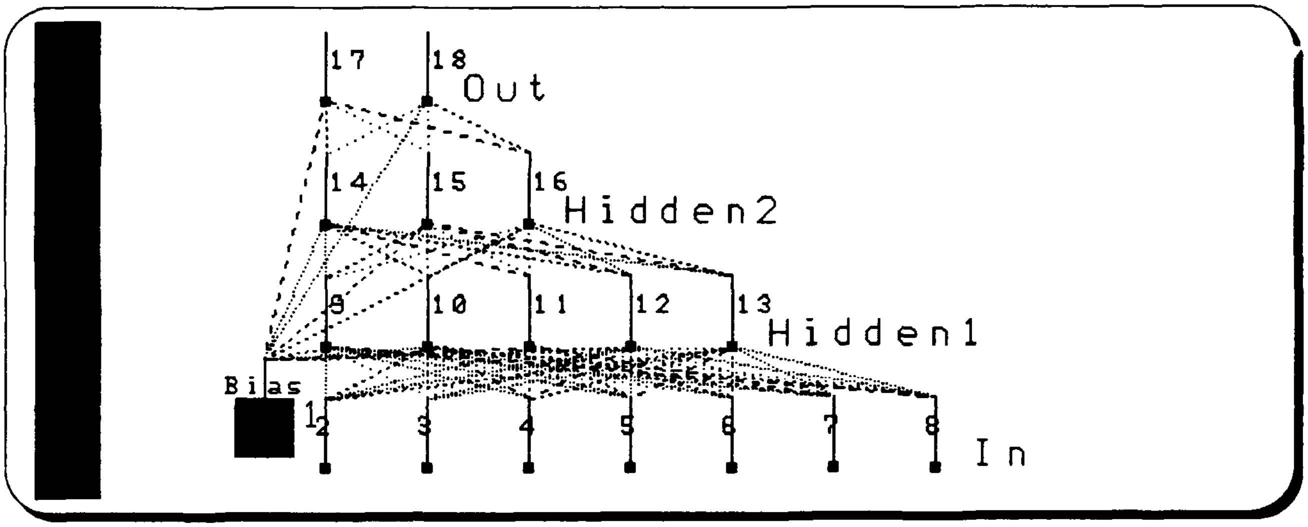


Figure 23. Experiment #4: Neural Network Topology.

EXPERIMENT FIVE: MODELS AND CODING

Experiment number five is the second of a series of three dimensional tracking experiments. The single eyelet of experiment number four is the basis and background for this experiment with an extension to three eyelets. Consider the three eyelets shown in Figure 24. Notice that there are three voltage plots, each corresponding to the output voltages of an eyelet. The point source is located at infinity while being viewed by the eyelets. The point source location is limited to positions that give signals on at least two eyelets.

There will be two experiments: experiment 5a and experiment 5b. The first experiment limits the point source location to between -125 milliradians and 125 milliradians. The point source offset shift is 125 milliradians for eyelets 1 and 2 as represented by the equations given on the following page. This optical configuration will give signals on only two eyelets. It is impossible to obtain non-zero signals on the output of three eyelets. In experiment 5b, the offset should be set to around +/- 50 milliradians such that there are regions of three eyelet stimulation. It will be interesting to note the difference in resolution where three eyelet overlap occurs compared to that of two eyelet overlap.

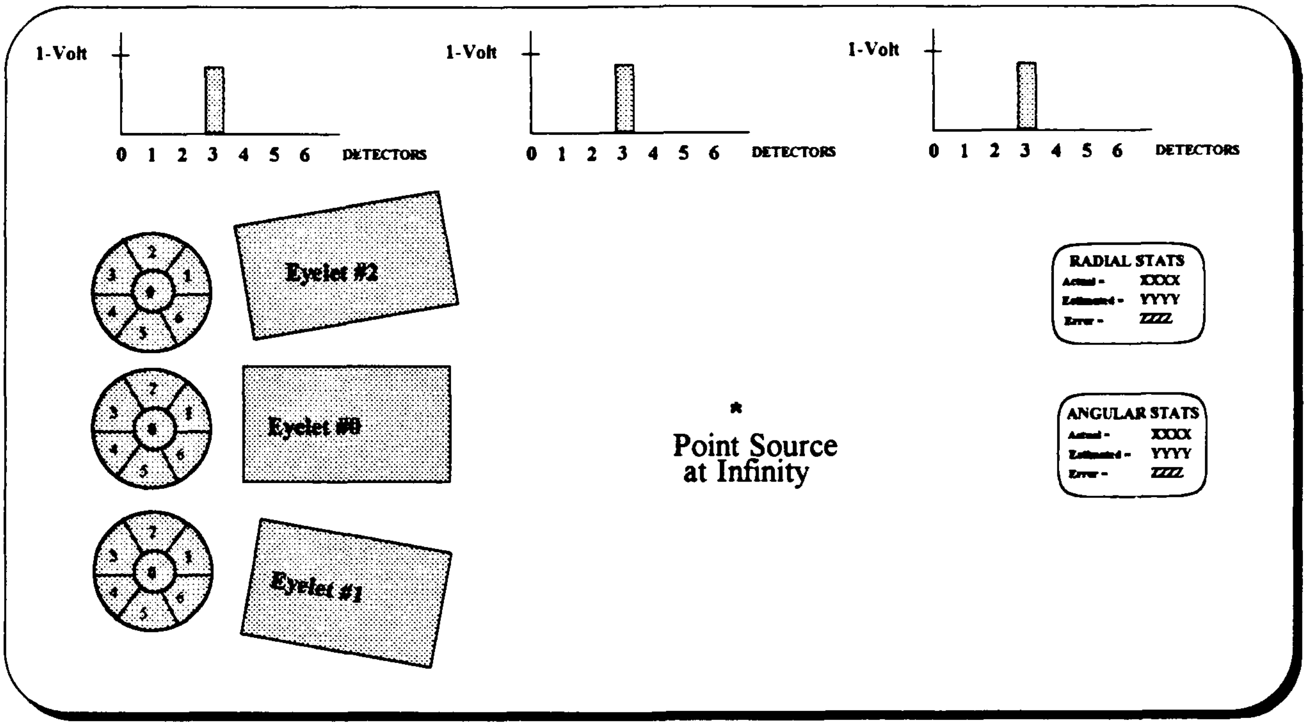


Figure 24. Experiment #5: Silicon Graphics Workstation Screen Graphics.

Let x be the ordinate angle in milliradians of the point source from the system optical axis and let y be the abscissa in milliradians. As defined in experiment number 4, the image point coordinate system for eyelet 0 is

$$\phi = \sqrt{x^2 + y^2}, \quad \theta = \tan^{-1}\left(\frac{y}{x}\right)$$

The coordinate system for eyelet 1 will be the same type of coordinate system as that for eyelet 0 with the exception of an abscissa shift upward.

$$\alpha = \sqrt{x^2 + (y - 125)^2}, \quad \beta = \tan^{-1}\left(\frac{y-125}{x}\right)$$

The eyelet 2 coordinate system is similar to eyelet 1 with a downward shift

$$\eta = \sqrt{x^2 + (y + 125)^2}, \quad \xi = \tan^{-1}\left(\frac{y+125}{x}\right)$$

The eyelet model equations are given on the following pages. Again, FEO anticipates using a similar neural network to those used in experiment's #1 through #4 except that the input will consist of 21 processing elements. Note also that not all 15 inputs are shown in figure 25 (this was done to enlarge the graphics to a readable scale). The identical analyses performed in the first three experiments will be implemented here also. The results of the simulation will be compared to the first three experiments as well as experiment #4. Additionally, the simulation interface will be identical to the first three experiments.

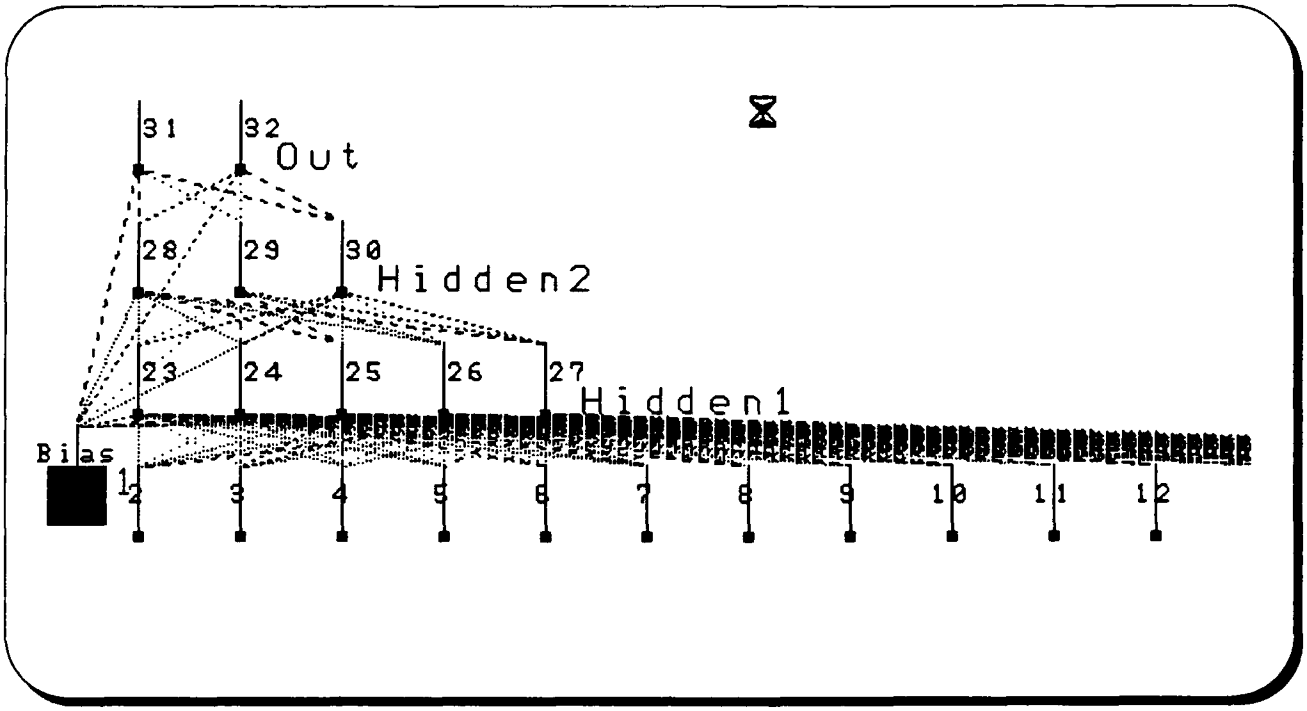


Figure 25. Experiment # 5: Neural Network Topology.

Eyelet 2 Equations

$$v_{20} = \sqrt{1 - (\frac{\eta}{125})^2}, \eta \leq 25 mrads$$

$$v_{21} = \sqrt{1 - (\frac{\eta}{125})^2}, 25 \leq \eta \leq 125 mrads, 0 \leq \xi < \frac{\pi}{3}$$

$$v_{22} = \sqrt{1 - (\frac{\eta}{125})^2}, 25 \leq \eta \leq 125 mrads, \frac{\pi}{3} \leq \xi < \frac{2\pi}{3}$$

$$v_{23} = \sqrt{1 - (\frac{\eta}{125})^2}, 25 \leq \eta \leq 125 mrads, \frac{2\pi}{3} \leq \xi < \pi$$

$$v_{24} = \sqrt{1 - (\frac{\eta}{125})^2}, 25 \leq \eta \leq 125 mrads, \pi \leq \xi < \frac{4\pi}{3}$$

$$v_{25} = \sqrt{1 - (\frac{\eta}{125})^2}, 25 \leq \eta \leq 125 mrads, \frac{4\pi}{3} \leq \xi < \frac{5\pi}{3}$$

$$v_{26} = \sqrt{1 - (\frac{\eta}{125})^2}, 25 \leq \eta \leq 125 mrads, \frac{5\pi}{3} \leq \xi < 2\pi$$

Eyelet 1 Equations

$$v_{10} = \sqrt{1 - (\frac{\alpha}{125})^2}, \alpha \leq 25 mrads$$

$$v_{11} = \sqrt{1 - (\frac{\alpha}{125})^2}, 25 \leq \alpha \leq 125 mrads, 0 \leq \beta < \frac{\pi}{3}$$

$$v_{12} = \sqrt{1 - (\frac{\alpha}{125})^2}, 25 \leq \alpha \leq 125 mrads, \frac{\pi}{3} \leq \beta < \frac{2\pi}{3}$$

$$v_{13} = \sqrt{1 - (\frac{\alpha}{125})^2}, 25 \leq \alpha \leq 125 mrads, \frac{2\pi}{3} \leq \beta < \pi$$

$$v_{14} = \sqrt{1 - (\frac{\alpha}{125})^2}, 25 \leq \alpha \leq 125 mrads, \pi \leq \beta < \frac{4\pi}{3}$$

$$v_{15} = \sqrt{1 - (\frac{\alpha}{125})^2}, 25 \leq \alpha \leq 125 mrads, \frac{4\pi}{3} \leq \beta < \frac{5\pi}{3}$$

$$v_{16} = \sqrt{1 - (\frac{\alpha}{125})^2}, 25 \leq \alpha \leq 125 mrads, \frac{5\pi}{3} \leq \beta < 2\pi$$

Eyelet 0 Equations

$$v_{00} = \sqrt{1 - (\frac{\phi}{125})^2}, \phi \le 25 mrads$$

$$v_{01} = \sqrt{1 - (\frac{\phi}{125})^2}, 25 \le \phi \le 125 mrads, 0 \le \theta < \frac{\pi}{3}$$

$$v_{02} = \sqrt{1 - (\frac{\phi}{125})^2}, 25 \le \phi \le 125 mrads, \frac{\pi}{3} \le \theta < \frac{2\pi}{3}$$

$$v_{03} = \sqrt{1 - (\frac{\phi}{125})^2}, 25 \le \phi \le 125 mrads, \frac{2\pi}{3} \le \theta < \pi$$

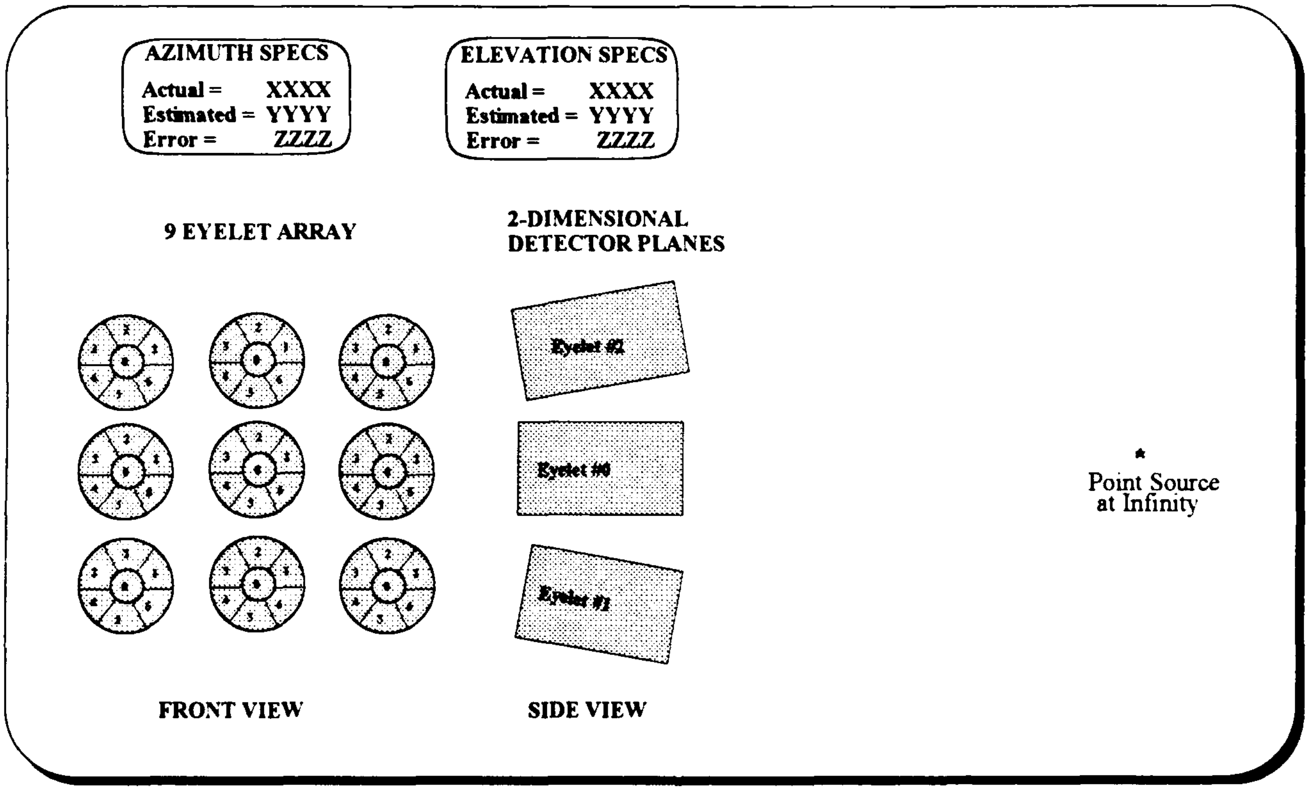
$$v_{04} = \sqrt{1 - (\frac{\phi}{125})^2}, 25 \le \phi \le 125 mrads, \pi \le \theta < \frac{4\pi}{3}$$

$$v_{05} = \sqrt{1 - (\frac{\phi}{125})^2}, 25 \le \phi \le 125 mrads, \frac{4\pi}{3} \le \theta < \frac{5\pi}{3}$$

$$v_{06} = \sqrt{1 - (\frac{\phi}{125})^2}, 25 \le \phi \le 125 mrads, \frac{5\pi}{3} \le \theta < 2\pi$$

EXPERIMENT SIX: MODELS AND CODING

In experiment number 6, a three by three array of eyelets is arranged in a manner as shown in figure 26. For this experiment a matrix-type notation will be used where i,j,k will denote lens position in the array, and detector number. The variable i will denote horizontal lens position where 0 is in the left, 1 is in the center, and 2 is to the right. The variable j will denote vertical lens position where 0 is up, 1 is in the center, and 2 is down. The variable k will denote detector number where 0 is in the lens center, 1 through 6 corresponds to 60 degree wedges counterclockwise at the $(0, 2\pi)$ boundary.



AZIMUTH SPECS
Actual = XXXX
Estimated = YYYY
Error = ZZZZ

ELEVATION SPECS
Actual = XXXX
Estimated = YYYY
Error = ZZZZ

Figure 26. Experiment #6: Silicon Graphics Workstation Graphics Screen.

For a given point in angular space, (x,y), where x and y are in milliradians, the neural network outputs are

$$\phi = \sqrt{x^2 + y^2}, \quad \theta = \tan^{-1}\left(\frac{y}{x}\right)$$

The detector voltages corresponding to any eyelet i,j are given on the following pages. The values in these equations are as follows:

x,y are the Cartesian coordinates of the target in milliradians
x_o, y_o are given in the table below for eyelet i,j in milliradians.

i	j	x_o	y_0
0	0	-125	-125
0	1	-125	0
0	2	-125	125
1	0	0	-125
1	1	0	0
1	2	0	125
2	0	125	-125
2	1	125	0
2	2	125	125

Note that eyelet 1,1 is now the reference eyelet for the system. A second experiment, 6b, will change the 125 milliradian values in the table to 50 milliradians.

The neural network for this experiment will change considerably from the previous five experiments. First, 63 input processing elements will be needed to receive the 63-detector outputs. Two middle layers will be used again with each layer increasing the number of processing elements. Initially, FEO will use a first middle layer with 25 processing elements and a second middle layer with 15 processing elements. Analysis of the results will proceed as before and the graphical interface will be similar to earlier experiments.

Eyelet i,j Equations

$$v_{i,j,0} = \sqrt{1-(\frac{\sqrt{(x-x_o)^2+(y-y_o)^2}}{125})^2}\,, \sqrt{(x-x_o)^2+(y-y_o)^2} \leq 25 mrads$$

$$v_{i,j,1} = \sqrt{1-(\frac{\sqrt{(x-x_o)^2+(y-y_o)^2}}{1\ \ 25})^2}\,, 25 \leq \sqrt{(x-x_o)^2+(y-y_o)^2} \leq 125 mrads\,, 0 \leq \mathrm{t\ an}^{-1}\frac{x-x_o}{y-y_o} < \frac{\pi}{3}$$

$$v_{i,j,2} = \sqrt{1-(\frac{\sqrt{(x-x_o)^2+(y-y_o)^2}}{1\ \ 25})^2}\,, 25 \leq \sqrt{(x-x_o)^2+(y-y_o)^2} \leq 125 mrads\,, \frac{\pi}{3} \leq \mathrm{t\ an}^{-1}\frac{x-x_o}{y-y_o} < \frac{2\pi}{3}$$

$$v_{i,j,3} = \sqrt{1-(\frac{\sqrt{(x-x_o)^2+(y-y_o)^2}}{1\ \ 25})^2}\,, 25 \leq \sqrt{(x-x_o)^2+(y-y_o)^2} \leq 125 mrads\,, \frac{2\pi}{3} \leq \mathrm{t\ an}^{-1}\frac{x-x_o}{y-y_o} < \pi$$

$$v_{i,j,4} = \sqrt{1-(\frac{\sqrt{(x-x_o)^2+(y-y_o)^2}}{1\ \ 25})^2}\,, 25 \leq \sqrt{(x-x_o)^2+(y-y_o)^2} \leq 125 mrads\,, \pi \leq \mathrm{t\ an}^{-1}\frac{x-x_o}{y-y_o} < \frac{4\pi}{3}$$

$$v_{i,j,5} = \sqrt{1-(\frac{\sqrt{(x-x_o)^2+(y-y_o)^2}}{1\ \ 25})^2}\,, 25 \leq \sqrt{(x-x_o)^2+(y-y_o)^2} \leq 125 mrads\quad \frac{4\pi}{3} \leq \mathrm{t\ an}^{-1}\frac{x-x_o}{y-y_o} < \frac{5\pi}{3}$$

$$v_{i,j,6} = \sqrt{1-(\frac{\sqrt{(x-x_o)^2+(y-y_o)^2}}{1\ \ 25})^2}\,, 25 \leq \sqrt{(x-x_o)^2+(y-y_o)^2} \leq 125 mrads\,, \frac{5\pi}{3} \leq \mathrm{t\ an}^{-1}\frac{x-x_o}{y-y_o} < 2\pi$$

EXPERIMENT SEVEN: MODELS AND CODING

Experiment #7, which uses the configuration of #6, will be implemented to study the tracking capabilities of the system. In considering this task, there are a large number of variables that can multiply very quickly. The outputs can be position, velocity, acceleration, and direction. The inputs can be detector voltages, last position, last velocity, potion two samples ago, velocity two samples ago, etc. With this complexity in mind, FEO's initial approach is to consider collinear (on a line) tracking. The cases will increment in the following manner:

(i) Constant Velocity

(ii) Constant Acceleration

(iii) Varied Acceleration

The bounds of the trajectories will include no target faster than 5 samples across the system FOV and no target slower than 100 samples across the system FOV.

Constant Velocity - trajectories will have constant velocity

Exp 7ia. NN Inputs: Detector voltages, last position

NN Outputs: target position, velocity, direction

Exp 7ib. NN Inputs: Detector voltages, last two positions

NN Outputs: target position, velocity, direction

Exp 7ic. NN Inputs: Detector voltages, last two positions, last velocity

NN Outputs: target position, velocity, direction

Exp 7id. NN Inputs: Detector voltages, last two positions, last velocity, last direction

NN Outputs: target position, velocity, direction

Constant Acceleration - trajectories will have constant acceleration

Exp 7iia. NN Inputs: Detector voltages, last position, last velocity

NN Outputs: target position, velocity, direction, acceleration

Exp 7iib. NN Inputs: Detector voltages, last two positions, last two velocities

NN Outputs: target position, velocity, direction, acceleration

Exp 7iic. NN Inputs: Detector voltages, last two positions, last two velocities, last dir.

NN Outputs: target position, velocity, direction, acceleration

Exp 7iid. NN Inputs: Detector voltages, last two positions, last two vel., last dir, accel.

NN Outputs: target position, velocity, direction, acceleration

Varied Acceleration - trajectories will have varied acceleration

Exp 7iiia. NN Inputs: Detector voltages, last position, last velocity

NN Outputs: target position, velocity, direction, acceleration

Exp 7iiib. NN Inputs: Detector voltages, last two positions, last two velocities

NN Outputs: target position, velocity, direction, acceleration

Exp 7iiic. NN Inputs: Detector voltages, last two positions, last two velocities, last dir.

NN Outputs: target position, velocity, direction, acceleration

Exp 7iiid. NN Inputs: Detector voltages, last two positions, last two vel., last dir, accel.

NN Outputs: target position, velocity, direction, acceleration

These approaches represent our initial consideration of the problem and may change as results are gathered. Note that this is only in a single tracking direction (no curves). It should also be noted that the optical models for these experiments are identical to those already implemented. It is the target trajectory models that must be changed. The bulk of the work is in the training of the neural networks.

CONCLUSIONS

To date, the results of the experiments have been better than expected considering the small number of detectors used. What was particularly impressive was the system's ability to provide reasonable passive tracking with only three eyelets (experiment #2). We expect the two-dimensional experiments, which we are currently training, will provide even better results. It is also interesting to note how informative the graphical simulation is when comparing it to the neural network error plots. The graphics will become more sophisticated as the project progresses. We have used the *basic* graphic primitives that Silicon Graphics offers during the early part of the project as we have concentrated largely on the optical models and neural network training.

We expect to have the fourth, fifth and sixth experiments on the IRIS Indigo and operational by mid-January. This will allow us one to two months to finish the last and most difficult seventh experiment which will determine the system's ability to estimate several tracking parameters including location, direction, velocity and acceleration. The last month of the project will be spent integrating all aspects of the system for the final product and final report (due March 15, 1993) as well as preparing for the Phase II submission.